

Problem E. Xor

Description

You are given 2^k numbers: $a_0, a_1, \dots, a_{(2^k-1)}$.

Define $s(x)$ as the following:

$$s(x) = \begin{cases} 0 & \text{if there are even bits of 1 in } x\text{'s binary representation} \\ 1 & \text{otherwise} \end{cases}$$

You need to support five types of operations:

- output $Xor_{i=l}^r s(i)a_i = s(l)a_l \text{ xor } s(l+1)a_{(l+1)} \text{ xor } \dots \text{ xor } s(r)a_r$.
- Assign $a_{(i \text{ xor } x)}$ to a_i simultaneously for each i in interval $[l, r]$.
- Assign a_i to $a_{(i \text{ xor } x)}$ simultaneously for each i in interval $[l, r]$.
- Assign $(a_i \text{ xor } x)$ to a_i simultaneously for each i in interval $[l, r]$.
- Assign y to a_x .

Xor here refers to bitwise operation *Exclusive Or*. In binary representation, the result in each position is 1 if the bits in two factors are different, and 0 if they are the same. For example,

$$5 \text{ xor } 3 = 6, \text{ which is } (101)_2 \text{ xor } (011)_2 = (110)_2$$

Input

The first line contains an integer T , indicating the number of test cases.

For each test case:

The first line contains an integer k ($0 \leq k \leq 17$).

The following line contains 2^k integers, the i -th integer is $a_{(i-1)}$ ($0 \leq a_i < 2^k$).

The next line contains an integer Q ($1 \leq Q < 2^k$), indicating the number of operations.

Then next Q lines, each line is one of:

- $1 \ l \ r$: output $Xor_{i=l}^r s(i)a_i$.

- $2\ x\ l\ r$: Assign $a_{(i\ xor\ x)}$ to a_i simultaneously for each i in interval $[l, r]$.
- $3\ x\ l\ r$: Assign a_i to $a_{(i\ xor\ x)}$ simultaneously for each i in interval $[l, r]$.
- $4\ x\ l\ r$: Assign $(a_i\ xor\ x)$ to a_i simultaneously for each i in interval $[l, r]$.
- $5\ x\ y$: Assign y to a_x .

It is guaranteed that $0 \leq x, y < 2^k$ and $0 \leq l \leq r < 2^k$.

Output

For each operation 1, output a line containing a integer, indicating the answer.

Sample Input

```
1
3
7 6 1 3 1 7 2 4
5
1 3 3
2 4 3 7
5 2 6
1 2 5
1 1 6
```

Sample Output

```
0
1
7
```